Aula 5

DevOps e Integração Contínua

Prof. Mauricio Antonio Ferste

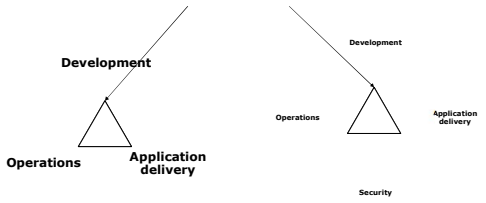
1

Conversa Inicial

2

DevOps com segurança

DevOps vs. DevSecOps



3

Princípios de segurança em DevOps

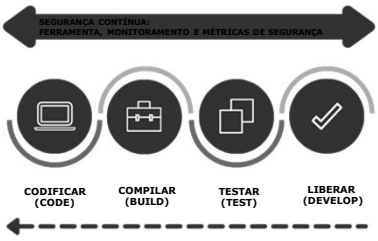
4

Princípios de segurança

- Integração e entrega contínua
- Implementação segura e ciclo de vida seguro
- Monitoramento contínuo
- Gestão de identidade e acesso (IAM)
- Resposta rápida a incidentes
- Testes automatizados

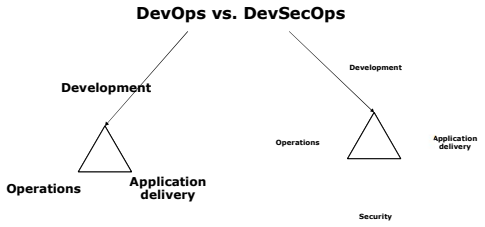
5

Contextualizando



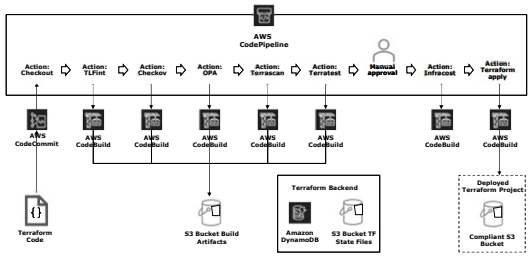
6

DevOps com segurança



7

Um exemplo – Terraform



8

Ferramentas para análise e gestão de riscos

9

Exemplos de riscos

- **Riscos de segurança: vulnerabilidades de segurança, acesso não autorizado, comprometimento de dados**
- **Riscos de disponibilidade: interrupções de serviço, falhas de hardware, falhas de software**
- **Riscos de desempenho: baixo desempenho, sobrecarga de recursos, latência alta**
- **Riscos de custo: custos excessivos, recursos subutilizados, recursos ociosos**

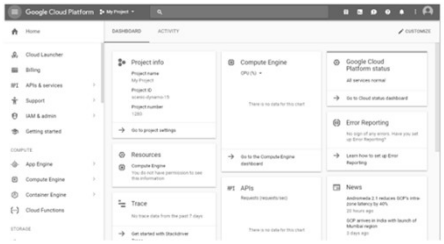
10

Vantagens de DevOps quanto a riscos

- **Automatização: automatizar muitas das tarefas envolvidas na análise e gestão de riscos, o que pode economizar tempo e recursos**
- **Precisão: ajudar a garantir que a análise e gestão de riscos sejam precisas e completas**
- **Visibilidade: fornecer uma visão abrangente dos riscos de infraestrutura, o que pode facilitar a tomada de decisões**

11

Exemplo de dashboard



12

Exemplo de dashboard criado com JSON e Terraform

https://newrelic.com/blog/how-to-relic/create-nr-dashboards-with-terraform-part-1

13/21

13

Exemplo de código para dashboard

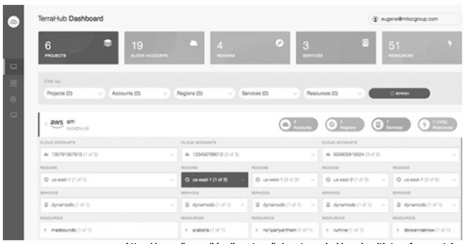
```
resource "newrelic_one_dashboard_json" "basic_dashboard" {
     json = file("${path.module}/dashboards/dashboard.json")
}

resource "newrelic_entity_tags" "basic_dashoard" {
    guid = newrelic_one_dashboard_json.basic_dashboard.guid
    tag {
              key    = "terraform"
              values = [true]
    }
}

output "basic_dashboard" {
     value = newrelic_one_dashboard_json.basic_dashboard.permali
}
```

https://newrelic.com/blog/how-to-relic/create-nr-dashboards-with-terraform-part-1

14/21

14

Outro exemplo de hub de controle



https://newrelic.com/blog/how-to-relic/create-nr-dashboards-with-terraform-part-1

15/21

15

Confiabilidade e continuidade em DevOps

16/21

16

O ciclo do medo

Fonte: Habbema, 2023.

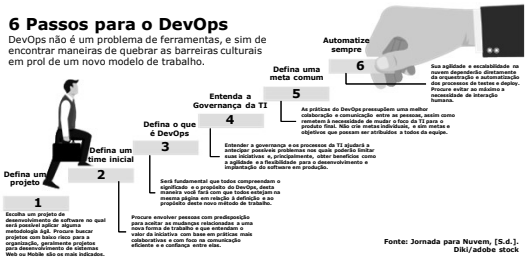
17/21

17

Tem de existir continuidade

6 Passos para o DevOps

DevOps não é um problema de ferramentas, e sim de encontrar maneiras de quebrar as barreiras culturais em prol de um novo modelo de trabalho.



Fonte: Jornada para Nuvem, [S.d.]. Diki/adobe stock

18/21

18

Autenticação e autorização

19

19

Práticas

- **Integrar a segurança desde o início: a segurança deve ser incorporada desde os estágios iniciais do desenvolvimento de software. Isso implica que as equipes de segurança colaborem com as equipes de desenvolvimento para identificar e mitigar riscos desde o início do processo**

20

20

Práticas

- **Manter a segurança após a implementação: a importância da segurança não termina com a implementação do software. As equipes de operações desempenham um papel vital, monitorando o software em busca de vulnerabilidades e ataques após a implantação**

21

21

Práticas

- **Adotar ferramentas de segurança automatizadas: ferramentas de segurança automatizadas desempenham um papel crucial em agilizar a verificação de segurança. Esse aspecto é particularmente relevante para equipes de DevOps que enfrentam a necessidade de realizar múltiplas revisões em um mesmo dia**

22

22

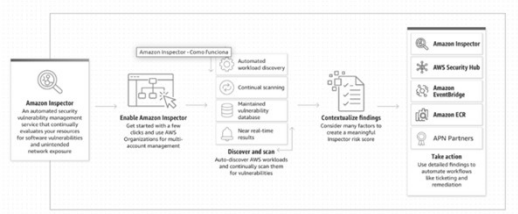
Práticas

- **Cultivar consciência de segurança: é essencial que toda a equipe de desenvolvimento de software esteja consciente da importância da segurança. Isso implica que cada membro da equipe esteja familiarizado com os riscos de segurança e compreenda como mitigá-los**

23

23

Exemplo de prática na Amazon – Inspector



24

24

Exemplo de prática na Amazon – Inspector

- **Exemplo de código de um provider para Inspector na AWS**

```
provider "aws" {
  region = "us-east-1"
}

resource "aws_inspector_assessment_target" "example" {
  name = "example-target"
}

resource "aws_inspector_assessment_template" "example" {
  name       = "example-template"
  duration   = 3600
  rules_package_arns = ["arn:aws:inspector:us-west-2:758058086616:rulespackage/0-9hgA516p"]
}
```

25

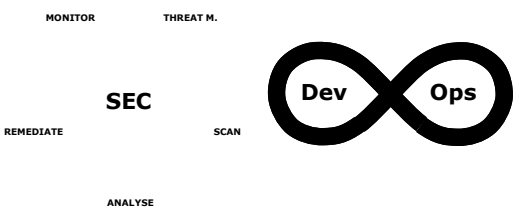
25

Segurança como código

26

26

Outro exemplo DevSecOps



https://www.targetso.com/2024/02/26/devsecops-2/

27

27

Exemplos de prática

- **Gerenciamento de credenciais: o primeiro passo crítico é assegurar o gerenciamento seguro de credenciais na AWS. Utilizar os serviços IAM (Identity and Access Management) é fundamental, criando roles e políticas que concedem acesso mínimo necessário. O Terraform pode ser configurado para utilizar essas credenciais de maneira segura, evitando a exposição indevida de chaves de acesso**

28

28

Exemplos de prática

- **Monitoramento e conformidade: implementar monitoramento contínuo para identificar e reagir a possíveis violações de segurança. Serviços como AWS config podem ser integrados para avaliação contínua da conformidade com políticas de segurança**

29

29

Exemplos de prática

- **Auditorias e revisões regulares: realizar auditorias regulares nas configurações do Terraform e nos recursos provisionados na AWS para identificar e corrigir potenciais lacunas de segurança. Essas revisões devem incluir análises de políticas de segurança, controle de acesso e configurações específicas de serviços AWS**

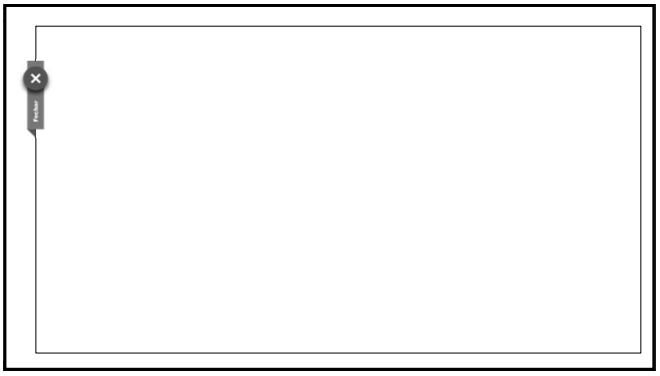
30

30

Exemplos de prática

- **Atualizações e patches automáticos: automatizar a aplicação de patches e atualizações de segurança para os recursos provisionados. Isso inclui a utilização de recursos como o AWS systems manager e AWS systems manager patch manager para garantir que os sistemas estejam sempre atualizados com as correções mais recentes**

31



32